Primitive recursion in finiteness spaces

Lionel Vaux

Primitive recursion in finiteness spaces

Lionel Vaux*

Laboratoire de Mathématiques de l'Université de Savoie
UFR SFA, Campus Scientifique, 73376 Le Bourget-du-Lac Cedex, FRANCE
E-mail: lionel.vaux@univ-savoie.fr

Abstract. We study iteration and recursion operators in the multiset relational model of linear logic and prove them finitary in the sense of the finiteness spaces recently introduced by Ehrhard. This provides a denotational semantics of Gödel's system T and paves the way for a systematic study of a large class of algorithms, following the ideas of Girard's quantitative semantics in a standard algebraic setting.

1 Introduction

The category $\underline{\mathrm{Fin}}$ of finiteness spaces and finitary relations was introduced by Ehrhard [1], refining the purely relational model of linear logic. A finiteness space is a set equipped with a finiteness structure, i.e. a particular set of subsets which are said to be finitary; and the model is such that the usual relational denotation of a proof in linear logic is always a finitary subset of its conclusion. By the usual co-Kleisli construction, this also provides a model of the simply typed lambda-calculus: the cartesian closed category $\underline{\mathrm{Fin}}_!$.

The main property of finiteness spaces is that the intersection of two finitary subsets of dual types is always finite. This feature allows to reformulate Girard's quantitative semantics in a standard algebraic setting, where morphisms interpreting typed λ-terms are analytic functions between the topological vector spaces generated by vectors with finitary supports. This provided the semantical foundations of Ehrhard-Regnier's differential λ-calculus [2] and motivated the general study of a differential extension of linear logic [3].

It is worth noticing that finiteness spaces can accomodate typed λ-calculi only: for instance, the relational semantics of fixpoint combinators is never finitary. The whole point of the finiteness construction is actually to reject infinite computations. Indeed, from a logical point of view, computation is cut elimination: the finiteness structure ensures the intermediate sets involved in the relational interpretation of a cut are all finite. In that sense, the finitary semantics is intrinsically typed.

Despite this restrictive design, Ehrhard proved that a limited form of recursion was available, by defining a finitary tail-recursive iteration operator. The main result of the present paper is that finiteness spaces can actually accomodate the usual notion of primitive recursion in models of the λ-calculus: $\underline{\mathrm{Fin}}_!$

* Supported by French ANR project CHOCO

admits a weak natural number object in the sense of [4,5], hence it is a model of the iterator variant of Gödel's system T. We moreover exhibit a recursion operator for this interpretation of the type of natural numbers and prove it is also finitary: finiteness spaces model recursion on all functionals of finite type.

Our construction interprets the type of natural numbers differently from that of Ehrhard. The latter relies on a peculiarity of tail-recursive iteration. Write Nat for the type of natural numbers, and let t, u and v be terms of types respectively Nat, $X \Rightarrow X$ and X. In the iteration step $\mathsf{J}\,(\mathsf{S}\,t)\,u\,v \rightsquigarrow \mathsf{J}\,t\,u\,(u\,v)$, no erasing nor duplication involves t, only the successor S is consumed: the tail-recursive iterator uses its integer argument linearly. The situation is very different in the case of the standard iterator: in $\mathsf{I}\,(\mathsf{S}\,t)\,u\,v \rightsquigarrow u\,(\mathsf{I}\,t\,u\,v)$, the term t is fed into the argument of u, which needs not be linear. In particular, if u is defined as a constant function (a weakening in linear logic terminology), then t is erased in the right hand side. In that case, the denotational semantics of $\mathsf{I}\,(\mathsf{S}\,t)\,u\,v$ must not depend on t. This forbids the successor S to be linear, because $\mathsf{S}\,t$ must produce a result to be distinguished from the null constant O, even without looking at t.[1] This phenomenon was already noted by Girard in his interpretation of system T in coherence spaces [6]. We adopt the solution he proposed in that setting, and interpret terms of type Nat by so-called *lazy* natural numbers.

Structure of the paper. We first provide a quick survey of the essential definitions and properties of finiteness spaces we shall use, and introduce the categories $\underline{\mathrm{Fin}}$ and $\underline{\mathrm{Fin}}_!$. We then make explicit the denotational semantics of typed λ-calculi in $\underline{\mathrm{Fin}}_!$, and present the finiteness space $\mathcal{N}_l$ of lazy natural numbers together with some finitary relations useful in later constructions. We prove that $\underline{\mathrm{Fin}}_!$ is a model of system T, by showing that lazy natural numbers form a weak natural number object: for all finiteness space $\mathcal{A}$, we define a finitary relation $\mathcal{I}_{\mathcal{A}}$ of type $\mathcal{N}_l \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}$, which obeys the equations of an iterator. We actually consider the stronger version of system T with a recursion operator and show that $\underline{\mathrm{Fin}}_!$ admits an internal recursor $\mathcal{R}_{\mathcal{A}}$ from which $\mathcal{I}_{\mathcal{A}}$ is derived. Hence the algorithmic expressivity of $\underline{\mathrm{Fin}}_!$ turns out to be quite rich. The relation $\mathcal{R}_{\mathcal{A}}$ is built as the fixpoint of a finitary relation $\mathcal{S}tep_{\mathcal{A}}$, but since the fixpoint operator is not itself finitary, we have to show the finitary approximants $(\mathcal{S}tep^n_{\mathcal{A}}(\emptyset))_{n\geq 0}$ enjoy additional properties to ensure $\mathcal{R}_{\mathcal{A}}$ is itself finitary.

2 Finiteness spaces

The construction of finiteness spaces follows a well known pattern. It is given by the following notion of orthogonality: $a \perp a'$ iff $a \cap a'$ is finite. Then one unrolls familiar definitions, as we do in the following paragraphs. For a more detailled presentation and proofs of the properties, the obvious reference is [1].

Let A be a set. Denote by $\mathfrak{P}\,(A)$ the powerset of A and by $\mathfrak{P}_{\mathrm{f}}\,(A)$ the set of all finite subsets of A. Let $\mathfrak{F} \subseteq \mathfrak{P}\,(A)$ any set of subsets of A. We define

[1] This "moral" argument might seem obscure to the reader not familiar with the relational or coherence semantics. It will be made formal later in the paper.

the pre-dual of $\mathfrak{F}$ in A as $\mathfrak{F}^{\perp_A} = \{a' \subseteq A;\ \forall a \in \mathfrak{F},\ a \cap a' \in \mathfrak{P}_{\mathrm{f}}(A)\}$. In general we will omit the subscript in the pre-dual notation and just write $\mathfrak{F}^{\perp}$. For all $\mathfrak{F} \subseteq \mathfrak{P}(A)$, we have the following immediate properties: $\mathfrak{P}_{\mathrm{f}}(A) \subseteq \mathfrak{F}^{\perp}$; $\mathfrak{F} \subseteq \mathfrak{F}^{\perp\perp}$; if $\mathfrak{G} \subseteq \mathfrak{F}$, $\mathfrak{F}^{\perp} \subseteq \mathfrak{G}^{\perp}$. By the last two, we get $\mathfrak{F}^{\perp} = \mathfrak{F}^{\perp\perp\perp}$. A finiteness structure on A is then a set $\mathfrak{F}$ of subsets of A such that $\mathfrak{F}^{\perp\perp} = \mathfrak{F}$.

A finiteness space is a dependant pair $\mathcal{A} = (|\mathcal{A}|, \mathfrak{F}(\mathcal{A}))$ where $|\mathcal{A}|$ is the underlying set (the web of $\mathcal{A}$) and $\mathfrak{F}(\mathcal{A})$ is a finiteness structure on $|\mathcal{A}|$. We then write $\mathcal{A}^{\perp}$ for the dual finiteness space: $\left|\mathcal{A}^{\perp}\right| = |\mathcal{A}|$ and $\mathfrak{F}\left(\mathcal{A}^{\perp}\right) = \mathfrak{F}(\mathcal{A})^{\perp}$. The elements of $\mathfrak{F}(\mathcal{A})$ are called the finitary subsets of $\mathcal{A}$.

Example 1. For all set A, $(A, \mathfrak{P}_{\mathrm{f}}(A))$ is a finiteness space and $(A, \mathfrak{P}_{\mathrm{f}}(A))^{\perp} = (A, \mathfrak{P}(A))$. In particular, each finite set A is the web of exactly one finiteness space: $(A, \mathfrak{P}_{\mathrm{f}}(A)) = (A, \mathfrak{P}(A))$. We introduce the following two: $\mathbf{0} = \mathbf{0}^{\perp} = (\emptyset, \{\emptyset\})$ and $\mathbf{1} = \mathbf{1}^{\perp} = (\{\emptyset\}, \{\emptyset, \{\emptyset\}\})$. We also introduce the finiteness space of natural numbers $\mathcal{N}$ by: $|\mathcal{N}| = \mathbf{N}$ and $a \in \mathfrak{F}(\mathcal{N})$ iff a is finite. We write $\mathcal{O} = \{0\} \in \mathfrak{F}(\mathcal{N})$.

Notice that $\mathfrak{F}$ is a finiteness structure iff it is of the form $\mathfrak{G}^{\perp}$. It follows that any finiteness structure $\mathfrak{F}$ is downwards closed for inclusion, and closed under finite unions and arbitrary intersections. Notice however that $\mathfrak{F}$ is not closed under directed unions in general: for all $k \in \mathbf{N}$, write $k{\downarrow} = \{j;\ j \leq k\} \in \mathfrak{F}(\mathcal{N})$; then $k{\downarrow} \subseteq k'{\downarrow}$ as soon as $k \leq k'$, but $\bigcup_{k \geq 0} k{\downarrow} = \mathbf{N} \notin \mathfrak{F}(\mathcal{N})$.

Multiplicatives. For all finiteness spaces $\mathcal{A}$ and $\mathcal{B}$, we define $\mathcal{A} \otimes \mathcal{B}$ by $|\mathcal{A} \otimes \mathcal{B}| = |\mathcal{A}| \times |\mathcal{B}|$ and $\mathfrak{F}(\mathcal{A} \otimes \mathcal{B}) = \{a \times b;\ a \in \mathfrak{F}(\mathcal{A}),\ b \in \mathfrak{F}(\mathcal{B})\}^{\perp\perp}$. It can be shown that $\mathfrak{F}(\mathcal{A} \otimes \mathcal{B}) = \{c \subseteq |\mathcal{A}| \times |\mathcal{B}|;\ c|_l \in \mathfrak{F}(\mathcal{A}),\ c|_r \in \mathfrak{F}(\mathcal{B})\}$, where $c|_l$ and $c|_r$ are the obvious projections.

Let $f \subseteq A \times B$ be a relation from A to B, we write $f^{\perp} = \{(\beta, \alpha);\ (\alpha, \beta) \in f\}$. For all $a \subseteq A$, we set $f \cdot a = \{\beta \in B;\ \exists \alpha \in a,\ (\alpha, \beta) \in f\}$. If moreover $g \subseteq B \times C$, we define $g \circ f = \{(\alpha, \gamma) \in A \times C;\ \exists \beta \in B,\ (\alpha, \beta) \in f \wedge (\beta, \gamma) \in g\}$. Then, setting $\mathcal{A} \multimap \mathcal{B} = \left(\mathcal{A} \otimes \mathcal{B}^{\perp}\right)^{\perp}$, $\mathfrak{F}(\mathcal{A} \multimap \mathcal{B}) \subseteq |\mathcal{A}| \times |\mathcal{B}|$ is characterized as follows:

$$\begin{aligned} f \in \mathfrak{F}(\mathcal{A} \multimap \mathcal{B}) \text{ iff } & \forall a \in \mathfrak{F}(\mathcal{A}),\ f \cdot a \in \mathfrak{F}(\mathcal{B}) \text{ and } \forall b \in \mathfrak{F}\left(\mathcal{B}^{\perp}\right),\ f^{\perp} \cdot b \in \mathfrak{F}\left(\mathcal{A}^{\perp}\right) \\ \text{iff } & \forall a \in \mathfrak{F}(\mathcal{A}),\ f \cdot a \in \mathfrak{F}(\mathcal{B}) \text{ and } \forall \beta \in |\mathcal{B}|,\ f^{\perp} \cdot \{\beta\} \in \mathfrak{F}\left(\mathcal{A}^{\perp}\right) \\ \text{iff } & \forall \alpha \in |\mathcal{A}|,\ f \cdot \{\alpha\} \in \mathfrak{F}(\mathcal{B}) \text{ and } \forall b \in \mathfrak{F}\left(\mathcal{B}^{\perp}\right),\ f^{\perp} \cdot b \in \mathfrak{F}\left(\mathcal{A}^{\perp}\right) \end{aligned}$$

The elements of $\mathfrak{F}(\mathcal{A} \multimap \mathcal{B})$ are called finitary relations from $\mathcal{A}$ to $\mathcal{B}$. By the previous characterization, the identity relation $id_{\mathcal{A}} = \{(\alpha, \alpha);\ \alpha \in |\mathcal{A}|\}$ is finitary, and the composition of two finitary relations is also finitary. One can thus define the category $\underline{\mathrm{Fin}}$ of finiteness spaces and finitary relations: the objects of $\underline{\mathrm{Fin}}$ are all finiteness spaces, and $\underline{\mathrm{Fin}}(\mathcal{A}, \mathcal{B}) = \mathfrak{F}(\mathcal{A} \multimap \mathcal{B})$. Equipped with the tensor product $\otimes$, $\underline{\mathrm{Fin}}$ is symmetric monoidal, with unit $\mathbf{1}$; it is monoidal closed by the definition of $\multimap$; it is $*$-autonomous by the obvious isomorphism between $\mathcal{A}^{\perp}$ and $\mathcal{A} \multimap \mathbf{1}$.

Example 2. Setting $\mathcal{S} = \{(k, k+1);\ k \in \mathbf{N}\}$ and $\mathcal{P} = \{(k+1, k);\ k \in \mathbf{N}\}$, we have $\mathcal{S}, \mathcal{P} \in \underline{\mathrm{Fin}}(\mathcal{N}, \mathcal{N})$ and $\mathcal{P} \circ \mathcal{S} = id_{\mathcal{N}}$.

Additives. We now introduce the cartesian structure of $\underline{\mathrm{Fin}}$. We define $\mathcal{A} \oplus \mathcal{B}$ by $|\mathcal{A} \oplus \mathcal{B}| = |\mathcal{A}| \uplus |\mathcal{B}|$ and $\mathfrak{F}(\mathcal{A} \oplus \mathcal{B}) = \{a \uplus b;\ a \in \mathfrak{F}(\mathcal{A}),\ b \in \mathfrak{F}(\mathcal{B})\}$ where $\uplus$ denotes the disjoint union of sets: $x \uplus y = (\{1\} \times x) \cup (\{2\} \times y)$. We have $(\mathcal{A} \oplus \mathcal{B})^{\perp} = \mathcal{A}^{\perp} \oplus \mathcal{B}^{\perp}$. The category $\underline{\mathrm{Fin}}$ is both cartesian and co-cartesian, with $\oplus$ being the product and co-product, and $\mathbf{0}$ the initial and terminal object. Projections are given by:

$$\lambda_{\mathcal{A},\mathcal{B}} = \{((1,\alpha),\alpha);\ \alpha \in |\mathcal{A}|\} \in \underline{\mathrm{Fin}}(\mathcal{A} \oplus \mathcal{B}, \mathcal{A})$$
$$\rho_{\mathcal{A},\mathcal{B}} = \{((2,\beta),\beta);\ \beta \in |\mathcal{B}|\} \in \underline{\mathrm{Fin}}(\mathcal{A} \oplus \mathcal{B}, \mathcal{B})$$

and if $f \in \underline{\mathrm{Fin}}(\mathcal{C}, \mathcal{A})$ and $g \in \underline{\mathrm{Fin}}(\mathcal{C}, \mathcal{B})$, pairing is given by:

$$\langle f, g\rangle = \{(\gamma,(1,\alpha));\ (\gamma,\alpha) \in f\} \cup \{(\gamma,(2,\beta));\ (\gamma,\beta) \in g\} \in \underline{\mathrm{Fin}}(\mathcal{C}, \mathcal{A} \oplus \mathcal{B}).$$

The unique morphism from $\mathcal{A}$ to $\mathbf{0}$ is the empty relation. The co-cartesian structure is obtained symmetrically.

Example 3. Write $\mathcal{O}^{\perp} = \{(0,\emptyset)\} \in \underline{\mathrm{Fin}}(\mathcal{N}, \mathbf{1})$. Then $\langle \mathcal{O}^{\perp}, \mathcal{P}\rangle = \{(0,(1,\emptyset))\} \cup \{(k+1,(2,k));\ k \in \mathbf{N}\} \in \underline{\mathrm{Fin}}(\mathcal{N}, \mathbf{1} \oplus \mathcal{N})$ is an isomorphism.

Exponentials. If A is a set, we denote by $\mathcal{M}_{\mathrm{fin}}(A)$ the set of all finite multisets of elements of A, and if $a \subseteq A$, we write $a^{!} = \mathcal{M}_{\mathrm{fin}}(a) \subseteq \mathcal{M}_{\mathrm{fin}}(A)$. If $\overline{\alpha} \in \mathcal{M}_{\mathrm{fin}}(A)$, we denote its support by $\mathrm{Supp}(\overline{\alpha}) \in \mathfrak{P}_{\mathrm{f}}(A)$. For all finiteness space $\mathcal{A}$, we define $!\mathcal{A}$ by: $|!\mathcal{A}| = \mathcal{M}_{\mathrm{fin}}(|\mathcal{A}|)$ and $\mathfrak{F}(!\mathcal{A}) = \left\{a^{!};\ a \in \mathfrak{F}(\mathcal{A})\right\}^{\perp\perp}$. It can be shown that $\mathfrak{F}(!\mathcal{A}) = \left\{\overline{a} \subseteq \mathcal{M}_{\mathrm{fin}}(|\mathcal{A}|);\ \bigcup_{\overline{\alpha} \in \overline{a}} \mathrm{Supp}(\overline{\alpha}) \in \mathfrak{F}(\mathcal{A})\right\}$. Then, for all $f \in \underline{\mathrm{Fin}}(\mathcal{A}, \mathcal{B})$, we set

$$!f = \{([\alpha_1,\ldots,\alpha_n],[\beta_1,\ldots,\beta_n]);\ \forall i,\ (\alpha_i,\beta_i) \in f\} \in \underline{\mathrm{Fin}}(!\mathcal{A}, !\mathcal{B}),$$

which defines a functor. Natural transformations $der_{\mathcal{A}} = \{([\alpha],\alpha);\ \alpha \in |\mathcal{A}|\} \in \underline{\mathrm{Fin}}(!\mathcal{A}, \mathcal{A})$ and $dig_{\mathcal{A}} = \{(\sum_{i=1}^{n} \overline{\alpha}_i, [\overline{\alpha}_1,\ldots,\overline{\alpha}_n]);\ \forall i,\ \overline{\alpha}_i \in |!\mathcal{A}|\}$ make this functor a comonad.

Example 4. We have isomorphisms $\{([],\emptyset)\} \in \underline{\mathrm{Fin}}(!\mathbf{0}, \mathbf{1})$ and

$$\left\{\left(\overline{\alpha}_l + \overline{\beta}_r, (\overline{\alpha},\overline{\beta})\right);\ (\overline{\alpha}_l,\overline{\alpha}) \in !\lambda_{\mathcal{A},\mathcal{B}} \wedge (\overline{\beta}_r,\overline{\beta}) \in !\rho_{\mathcal{A},\mathcal{B}}\right\} \in \underline{\mathrm{Fin}}(!(\mathcal{A} \oplus \mathcal{B}), !\mathcal{A} \otimes !\mathcal{B}).$$

More generally, we have $!(\mathcal{A}_1 \oplus \cdots \oplus \mathcal{A}_n) \cong !\mathcal{A}_1 \otimes \cdots \otimes !\mathcal{A}_n$.

All the usual structure of models of propositional classical linear logic is now introduced: one can check that $\underline{\mathrm{Fin}}$ is a new-Seely category, following the terminology of [7].

The co-Kleisli construction. Since the structure we have just described provides a denotational model of linear logic, the co-Kleisli construction gives rise to a cartesian closed category, which we denote by $\underline{\mathrm{Fin}}_{!}$: objects are finiteness spaces,

$$\frac{}{\Gamma, x : A, \Delta \vdash x : A}\ (\text{Var}) \qquad \frac{}{\Gamma \vdash \langle\rangle : \top}\ (\text{Unit}) \qquad \frac{a \in \mathfrak{C}_A}{\Gamma \vdash a : A}\ (\text{Const})$$

$$\frac{\Gamma, x : A \vdash s : B}{\Gamma \vdash \lambda x\, s : A \to B}\ (\text{Abs}) \qquad \frac{\Gamma \vdash s : A \to B \qquad \Gamma \vdash t : A}{\Gamma \vdash s\, t : B}\ (\text{App})$$

$$\frac{\Gamma \vdash s : A \qquad \Gamma \vdash t : B}{\Gamma \vdash \langle s, t\rangle : A \times B}\ (\text{Pair}) \qquad \frac{\Gamma \vdash s : A \times B}{\Gamma \vdash \pi_l\, s : A}\ (\text{Left}) \qquad \frac{\Gamma \vdash s : A \times B}{\Gamma \vdash \pi_r\, s : B}\ (\text{Right})$$

Fig. 1. Rules of typed λ-calculi with products

and $\underline{\text{Fin}}_!(\mathcal{A}, \mathcal{B}) = \mathfrak{F}(\mathcal{A} \Rightarrow \mathcal{B})$, following Girard's translation of intuitionistic implication $\mathcal{A} \Rightarrow \mathcal{B} = !\mathcal{A} \multimap \mathcal{B}$. The identity morphism of $\underline{\text{Fin}}_!(\mathcal{A}, \mathcal{A})$ is just $der_{\mathcal{A}}$. If $f \in !\mathcal{A} \multimap \mathcal{B}$, the promotion of f is

$$f^! = !f \circ dig_{\mathcal{A}} = \left\{ \left(\sum_{i=1}^{n} \overline{\alpha}_i, [\beta_1, \ldots, \beta_n] \right) ;\ \forall i,\ (\overline{\alpha}_i, \beta_i) \in f \right\} \in !\mathcal{A} \multimap !\mathcal{B}.$$

If $f \in \mathcal{A} \Rightarrow \mathcal{B}$ and $g \in \mathcal{B} \Rightarrow \mathcal{C}$, their composition in $\underline{\text{Fin}}_!$ is then $g \bullet f = g \circ f^!$.

Remark 1. Recalling the characterizations of $\mathcal{A} \multimap \mathcal{B}$ and $!\mathcal{A}$, we get that $f \in \mathfrak{F}(\mathcal{A} \Rightarrow \mathcal{B})$ iff $\forall a \in \mathfrak{F}(\mathcal{A})$, $f \cdot a^! \in \mathfrak{F}(\mathcal{B})$ and $\forall \beta \in |\mathcal{B}|$, $(f^{\perp} \cdot \{\beta\}) \cap a^!$ is finite.

The cartesian product of $\underline{\text{Fin}}_!$ is of course $\oplus$, with projections $\pi_l^{\mathcal{A},\mathcal{B}} = \lambda_{\mathcal{A},\mathcal{B}} \circ der_{\mathcal{A}\oplus\mathcal{B}}$ and $\pi_r^{\mathcal{A},\mathcal{B}} = \rho_{\mathcal{A},\mathcal{B}} \circ der_{\mathcal{A}\oplus\mathcal{B}}$; pairing is the same as in $\underline{\text{Fin}}$. The isomorphism $(\mathcal{A}_1 \oplus \cdots \oplus \mathcal{A}_n) \Rightarrow \mathcal{B} \cong \mathcal{A}_1 \Rightarrow \cdots \Rightarrow \mathcal{A}_n \Rightarrow \mathcal{A}$ deduced from Example 4 subsumes the adjunction $\underline{\text{Fin}}_!(\mathcal{A} \oplus \mathcal{B}, \mathcal{C}) \cong \underline{\text{Fin}}_!(\mathcal{A}, \mathcal{B} \Rightarrow \mathcal{C})$. As a particular case, $\mathbf{0} \Rightarrow \mathcal{A} \cong \mathcal{A}$ and we can identify any morphism $f \in \mathbf{0} \Rightarrow \mathcal{A}$ with the finitary subset $f \cdot \{[]\} \in \mathfrak{F}(\mathcal{A})$.

3 Semantics of typed λ-calculi

In this section, we give an explicit description of the interpretation in $\underline{\text{Fin}}_!$ of the basic constructions of typed λ-calculi with products.

Typed λ-calculi. Let be given a collection $\mathfrak{A}$ of atomic types and consider the type expressions given by:

$$A, B ::= X \mid A \to B \mid A \times B \mid \top$$

where X ranges over $\mathfrak{A}$. For each type A, there are countably many formal variables of that type, and also a collection $\mathfrak{C}_A$ of term constants of that type. Then term expressions are given by:

$$s, t ::= x \mid a \mid \lambda x\, s \mid s\, t \mid \langle s, t\rangle \mid \pi_l\, s \mid \pi_r\, s \mid \langle\rangle$$

$$\frac{}{\Gamma^{[]}, x^{[\alpha]} : A, \Delta^{[]} \vdash x^{\alpha} : A}\ [\![\text{Var}]\!] \qquad \frac{}{\Gamma^{[]} \vdash \langle\rangle^{\emptyset} : \top}\ [\![\text{Unit}]\!] \qquad \frac{a \in \mathfrak{C}_A \quad \alpha \in [\![a]\!]}{\Gamma^{[]} \vdash a^{\alpha} : A}\ [\![\text{Const}]\!]$$

$$\frac{\Gamma, x^{\overline{\alpha}} : A \vdash s^{\beta} : B}{\Gamma \vdash \lambda x\, s^{(\overline{\alpha},\beta)} : A \to B}\ [\![\text{Abs}]\!]$$

$$\frac{\Gamma_0 \vdash s^{([\alpha_1,\ldots,\alpha_k],\beta)} : A \to B \qquad \Gamma_1 \vdash t^{\alpha_1} : A \quad \cdots \quad \Gamma_k \vdash t^{\alpha_k} : A}{\sum_{j=0}^{k} \Gamma_j \vdash s\, t^{\beta} : B}\ [\![\text{App}]\!]$$

$$\frac{\Gamma \vdash s^{\alpha} : A}{\Gamma \vdash \langle s, t\rangle^{(1,\alpha)} : A \times B}\ [\![\text{Pair}_l]\!] \qquad \frac{\Gamma \vdash t^{\beta} : B}{\Gamma \vdash \langle s, t\rangle^{(2,\beta)} : A \times B}\ [\![\text{Pair}_r]\!]$$

$$\frac{\Gamma \vdash s^{(1,\alpha)} : A \times B}{\Gamma \vdash \pi_l\, s^{\alpha} : A}\ [\![\text{Left}]\!] \qquad \frac{\Gamma \vdash s^{(2,\beta)} : A \times B}{\Gamma \vdash \pi_r\, s^{\beta} : B}\ [\![\text{Right}]\!]$$

Fig. 2. Computing points in the relational semantics

and we define free and bound variables as usual. We denote by $s\,[x := t]$ the substitution of t for x in s.

A context Γ is a finite list $(x_1 : A_1, \ldots, x_n : A_n)$ where the x_i's are pairwise distinct variables and A_i is the type of x_i. A typing judgement is an expression $\Gamma \vdash s : A$ derived from the rules in Figure 1: we then say term s is of type A in context Γ. Clearly, if a term s is typable, then its type is uniquely determined, say A, and then $\Gamma \vdash s : A$ iff Γ contains the free variables of s.

The operational semantics of a typed λ-calculus is given by a contextual equivalence relation $\simeq$ on typed terms: if $s \simeq t$, then s and t have the same type, say A; we then write $\Gamma \vdash s \simeq t : A$ whenever Γ contains the free variables of s and t. In general, we will give $\simeq$ as the reflexive, symmetric and transitive closure of a contextual relation $>$ on typed terms. We define $>_0$ as the least one such that: $\pi_l\,\langle s, t\rangle >_0 s$, $\pi_r\,\langle s, t\rangle >_0 t$ and $(\lambda x\, s)\, t >_0 s\,[x := t]$ (with the obvious assumptions ensuring typability). And we write $\simeq_0$ for the corresponding equivalence.

Relational interpretation and finiteness property. The interpretation of such a calculus in $\underline{\text{Fin}}_!$ goes as follows. First on types: assume a finiteness space $[\![X]\!]$ is given for each base type X; then we interpret type constructions by $[\![A \to B]\!] = [\![A]\!] \Rightarrow [\![B]\!]$, $[\![A \times B]\!] = [\![A]\!] \oplus [\![B]\!]$ and $[\![\top]\!] = \mathbf{0}$. Further assume that with every constant $a \in \mathfrak{C}_A$ is associated a finitary subset $[\![a]\!] \in \mathfrak{F}([\![A]\!])$: we define the semantics of a derivable typing judgement $x_1 : A_1, \ldots, x_n : A_n \vdash s : A$, as a finitary relation $[\![s]\!]_{x_1:A_1,\ldots,x_n:A_n} \in \mathfrak{F}([\![A_1]\!] \Rightarrow \cdots \Rightarrow [\![A_n]\!] \Rightarrow [\![A]\!])$.

We first introduce the deductive system of Figure 2. In this system, derivable judgements are semantic annotations of typing judgements:

$$x_1^{\overline{\alpha}_1} : A_1, \ldots, x_n^{\overline{\alpha}_n} : A_n \vdash s^{\alpha} : A \quad \text{stands for} \quad (\overline{\alpha}_1, \ldots, \overline{\alpha}_n, \alpha) \in [\![s]\!]_{x_1:A_1,\ldots,x_n:A_n}$$

where each $\overline{\alpha}_i \in \mathcal{M}_{\text{fin}}(|[\![A_i]\!]|)$ and $\alpha \in |[\![A]\!]|$. In rules [Var], [Unit] and [Const], $\Gamma^{[]}$ denotes an annotated context of the form $x_1^{[]} : A_1, \ldots, x_n^{[]} : A_n$. In rule [App], the sum of annotated contexts is defined pointwise: if $\Gamma = x_1^{\overline{\alpha}_1} : A_1, \ldots, x_n^{\overline{\alpha}_n} : A_n$ and $\Gamma' = x_1^{\overline{\alpha}'_1} : A_1, \ldots, x_n^{\overline{\alpha}'_n} : A_n$, we set $\Gamma + \Gamma' = x_1^{\overline{\alpha}_1 + \overline{\alpha}'_1} : A_1, \ldots, x_n^{\overline{\alpha}_n + \overline{\alpha}'_n} : A_n$. Finally we define the semantics of a term as the set of its annotations:

$$[\![s]\!]_{x_1:A_1,\ldots,x_n:A_n} = \left\{ (\overline{\alpha}_1, \ldots, \overline{\alpha}_n, \alpha);\ x_1^{\overline{\alpha}_1} : A_1, \ldots, x_n^{\overline{\alpha}_n} : A_n \vdash s^{\alpha} : A \right\}.$$

It remains to prove that $[\![s]\!]_{x_1:A_1,\ldots,x_n:A_n} \in \mathfrak{F}([\![A_1]\!] \Rightarrow \cdots \Rightarrow [\![A_n]\!] \Rightarrow [\![A]\!])$.

Remark 2. The rules of Figure 2 are actually those of the relational semantics: this is the semantics of typed λ-calculi in the cartesian closed category $\underline{\text{Rel}}_!$, deduced from the relational model of linear logic in the category $\underline{\text{Rel}}$ of sets and relations (see the Appendix A from [1] for instance). Notice indeed that the definitions of the natural transformations of $\underline{\text{Fin}}$ (and the definitions of the morphism component of the associated functors) that make it a model of classical linear logic, as presented in section 2, are free from any reference to the finiteness structure: they are just introduced as relations on webs, and it simply turns out they are finitary, whatever finiteness structure we consider on the underlying objects. The only restriction we make in the interpretation of λ-calculi is that $[\![a]\!] \in \mathfrak{F}([\![A]\!])$ for all $a \in \mathfrak{C}_A$.

Theorem 1 (Finiteness). *The semantics of a typed term is finitary: if* $x_1 : A_1, \ldots, x_n : A_n \vdash s : A$ *then* $[\![s]\!]_{x_1:A_1,\ldots,x_n:A_n} \in \mathfrak{F}([\![A_1]\!] \Rightarrow \cdots \Rightarrow [\![A_n]\!] \Rightarrow [\![A]\!])$.

Proof This holds directly because the semantics we have introduced is just rephrasing the usual interpretation of typed λ-calculi in a cartesian closed category [5], in the particular case of $\underline{\text{Fin}}_!$. Notably: rules [Unit], [Pair$_l$], [Pair$_r$], [Left] and [Right] match the definition of the cartesian structure of $\underline{\text{Fin}}_!$; rules [Abs] and [App] match the adjunction $\underline{\text{Fin}}_!(\Gamma \oplus \mathcal{A}, \mathcal{B}) \cong \underline{\text{Fin}}_!(\Gamma, \mathcal{A} \Rightarrow \mathcal{B})$ and the definition of composition in $\underline{\text{Fin}}_!$.

For the reader not familiar with this construction, a direct proof is also possible by induction on typing derivations. The only non-trivial case is that of [App], which essentially boils down to Remark 1. □

The interpretation of terms we have just defined is of course a model of $\simeq_0$ whatever the choice of base types and constants.

Theorem 2 (Invariance). *If* $\Gamma \vdash s \simeq_0 t : A$ *then* $[\![s]\!]_\Gamma = [\![t]\!]_\Gamma$.

Proof Again, this follows directly from the fact that we used the standard translation of typed λ-calculi in cartesian closed categories. A direct proof is also easy:

- first show by induction on s that, if $\Gamma_0, x : A^{[\alpha_1,\ldots,\alpha_k]}, \Delta_0 \vdash s^{\beta} : B$, and, for all $j \in \{1, \ldots, k\}$, $\Gamma_j, \Delta_j \vdash t^{\alpha_j} : A$, then $\sum_{j=0}^{k} \Gamma_j, \sum_{j=0}^{k} \Delta_j \vdash s\,[x := t]^{\beta} : B$;
- then check that for a basic reduction $\Gamma \vdash s >_0 t : A$, $(\overline{\alpha}_1, \ldots, \overline{\alpha}_n, \alpha) \in [\![s]\!]_\Gamma$ iff $(\overline{\alpha}_1, \ldots, \overline{\alpha}_n, \alpha) \in [\![t]\!]_\Gamma$, which is direct by the rules of Figure 2 and the previous result on substitution;

– conclude by contextuality of the semantics.

Notice that the invariance of the semantics has nothing to do with the finiteness structure: the result is actually a property of the relational interpretation only. □

Examples. Let us consider some particular calculi. Pure typed λ-calculi are those with no additional constant or conversion rule: fix a set $\mathfrak{A}$ of atomic types, and write $\Lambda_0^{\mathfrak{A}}$ for the calculus where $\mathfrak{C}_A = \emptyset$ for all A, and $s \simeq t$ iff $s \simeq_0 t$. This is the most basic case and we have just shown that finiteness spaces model $\simeq_0$. Be aware that if we introduce no atomic type, then the semantics is actually trivial: in $\Lambda_0^{\emptyset}$, all types are interpreted by $\mathbf{0}$ and all terms by the empty set.

By contrast, we can consider the internal language $\Lambda_{\underline{\mathrm{Fin}}}$ of $\underline{\mathrm{Fin}}_!$ in which all finitary relations can be described: fix $\mathfrak{A}$ as the collection of all finiteness spaces and $\mathfrak{C}_A = \mathfrak{F}(\llbracket A \rrbracket)$. Then set $s \simeq_{\underline{\mathrm{Fin}}} t$ iff $\llbracket s \rrbracket_\Gamma = \llbracket t \rrbracket_\Gamma$, for any suitable Γ. The point in defining such a monstrous language is to enable very natural notations for finitary relations: in general, we will identify closed terms in $\Lambda_{\underline{\mathrm{Fin}}}$ with the relations they denote in the empty context. In that case, we might make explicit the type of bound variables. For instance, we write $\lambda x^{\mathcal{A}}\, x = der_{\mathcal{A}}$; and if $f \in \underline{\mathrm{Fin}}_!(\mathcal{A}, \mathcal{B})$ and $g \in \underline{\mathrm{Fin}}_!(\mathcal{B}, \mathcal{C})$, we have $\lambda x^{\mathcal{A}}\, (g\,(f\,x)) = g \bullet f \in \underline{\mathrm{Fin}}_!(\mathcal{A}, \mathcal{B})$.

Before we address the main subject of the paper, system T, let's just review some easy examples of usual λ-calculi constructions that can be modelled in $\underline{\mathrm{Fin}}_!$. First, being a cartesian closed category, $\underline{\mathrm{Fin}}_!$ is actually an model of pure typed λ-calculi with extensionality, surjective pairing and terminal object: Theorem 2 still holds if we add the reductions $\lambda x\,(u\,x) >_0 u$, $\langle \pi_l\, s, \pi_r\, s \rangle >_0 s$ and $v >_0 \langle\rangle$ as soon as x is not free in u and v has type $\top$ (notice, however, that in that case $>_0$ is no longer confluent [5]).

We can also extend the language with particular base types and constants. For instance, we can introduce base type Bool together with constants T and F of type Bool, and D_A of type $\mathsf{Bool} \to A \to A \to A$, with the additional reductions $\mathsf{D}\,\mathsf{T}\,s\,t > s$ and $\mathsf{D}\,\mathsf{F}\,s\,t > t$ (we will in general omit the type subscript of such parametered constants, with the obvious hypotheses on typability) and fix interpretations as follows: write $\mathcal{B} = \mathbf{1} \oplus \mathbf{1}$ and write the only two elements of $|\mathcal{B}|$ as tt and ff; then let $\llbracket \mathsf{Bool} \rrbracket = \mathcal{B}$, $\llbracket \mathsf{T} \rrbracket = \mathcal{T} = \{\mathit{tt}\}$, $\llbracket \mathsf{F} \rrbracket = \mathcal{F} = \{\mathit{ff}\}$ and $\llbracket \mathsf{D}_A \rrbracket = \mathcal{D}_{\llbracket A \rrbracket} = \{([\mathit{tt}], [\alpha], [], \alpha);\ \alpha \in \llbracket A \rrbracket\} \cup \{([\mathit{ff}], [], [\alpha], \alpha);\ \alpha \in \llbracket A \rrbracket\}$. That these interpretations are finitary should be clear. Then one retains that $\Gamma \vdash s \simeq t : A$ implies $\llbracket s \rrbracket_\Gamma = \llbracket t \rrbracket_\Gamma$.

System T. The main contribution of the present paper is to establish that $\underline{\mathrm{Fin}}_!$ models Gödel's system T, which can be presented in various ways. The iterator version of system T is the typed λ-calculus with an atomic type Nat of natural numbers, and constants O of type Nat, S of type $\mathsf{Nat} \to \mathsf{Nat}$ and for all type A, I_A of type $\mathsf{Nat} \to (A \to A) \to A \to A$ and subject to the following additional conversions: $\mathsf{I}\,(\mathsf{O})\,u\,v > v$ and $\mathsf{I}\,(\mathsf{S}\,t)\,u\,v > u\,(\mathsf{I}\,t\,u\,v)$. The recursor variant is similar, but the iterator is replaced with R_A of type $\mathsf{Nat} \to (\mathsf{Nat} \to A \to A) \to A \to A$ subject to conversions $\mathsf{R}\,(\mathsf{O})\,u\,v > v$ and $\mathsf{R}\,(\mathsf{S}\,t)\,u\,v > u\,t\,(\mathsf{R}\,t\,u\,v)$. Yet

another possible system is obtained with tail recursive iteration: take J_A of type $\mathsf{Nat} \to (A \to A) \to A \to A$ and let $\mathsf{J}\,(\mathsf{S}\,t)\,u\,v > \mathsf{J}\,t\,u\,(u\,v)$.

Those systems allow to represent exactly the same functions on the set of natural numbers, where the number n is obviously denoted by $\mathsf{S}^n\,\mathsf{O}$: this is the consequence of a normalization theorem (see [6]). Notice in particular that, when applied to a canonical integer, I and J coincide: $\mathsf{I}\,(\mathsf{S}^n\,\mathsf{O})uv \simeq \mathsf{J}\,(\mathsf{S}^n\,\mathsf{O})uv \simeq u^n\,v$; this does not hold for all term of type Nat, however. Also, we can define a recursor using iteration and products with the standard encoding $\mathsf{rec} = \lambda x\,\lambda y\,\lambda z\,\pi_l\,(\mathsf{I}\,x\,\langle z, \mathsf{O}\rangle\,\lambda w\,\langle y\,(\pi_r\,w)\,(\pi_l\,w), \mathsf{S}\,(\pi_r\,w)\rangle)$, and we get $\mathsf{rec}\,(\mathsf{S}^n\,\mathsf{O})uv \simeq \mathsf{R}\,(\mathsf{S}^n\,\mathsf{O})uv$. The idea is to reconstruct the integer argument on the fly. Again, this encoding is valid only by values and $\mathsf{rec}\,(\mathsf{S}\,t)\,u\,v \simeq u\,t\,(\mathsf{rec}\,t\,u\,v)$ holds only if we suppose t is of the form $\mathsf{S}^n\,\mathsf{O}$ (notice we could as well use J in the definition of rec). Only the encoding of the iterator by $\mathsf{iter} = \lambda x\,\lambda y\,\lambda z\,(\mathsf{R}\,x\,(\lambda x'\,y)\,z)$ is extensionally valid: $\mathsf{iter}\,\mathsf{O}\,u\,v \simeq v$ and $\mathsf{iter}\,(\mathsf{S}\,t)\;u\,v \simeq u\,(\mathsf{iter}\,t\,u\,v)$.

The fact that in general the encoding of one system into the other holds only on values indicate that their respective algorithmic properties may differ. And these differences will appear in the semantics. Recall for instance the discussion in our introduction: J uses its integer argument linearly. This enabled Ehrhard to define a semantics of iteration, with $[\![\mathsf{Nat}]\!] = \mathcal{N} = (\mathbf{N}, \mathfrak{P}_{\mathrm{f}}\,(\mathbf{N}))$, $[\![\mathsf{O}]\!] = \mathcal{O} = \{0\}$ and $[\![\mathsf{S}]\!] = \mathcal{S}_! = \{([n]\,, n+1);\ n \in \mathbf{N}\}$. Such an interpretation of natural numbers, however, fails to provide a semantics of I or R.

Lemma 1. *Assume $[\![\mathsf{Nat}]\!] = \mathcal{N}$, $[\![\mathsf{O}]\!] = \mathcal{O}$ and $[\![\mathsf{S}]\!] = \mathcal{S}_!$, and let A be any type such that $[\![A]\!] \neq \mathbf{0}$. Then there is no $\mathcal{I}_A \in \mathfrak{F}\,(\mathcal{N} \Rightarrow ([\![A]\!] \Rightarrow [\![A]\!]) \Rightarrow [\![A]\!] \Rightarrow [\![A]\!])$ such that, setting $[\![\mathsf{I}_A]\!] = \mathcal{I}_A$, we obtain $[\![\mathsf{I}\,\mathsf{O}\,u\,v]\!]_\Gamma = [\![v]\!]_\Gamma$ and $[\![\mathsf{I}\,(\mathsf{S}\,t)\;u\,v]\!]_\Gamma = [\![u\,(\mathsf{I}\,t\,u\,v)]\!]_\Gamma$ as soon as $\Gamma \vdash t : \mathsf{Nat}$, $\Gamma \vdash u : A \to A$ and $\Gamma \vdash v : A$.*

Proof By contradiction, assume the above equations hold. We get

$$[\![\mathsf{I}\,(\mathsf{S}\,x)\;(\lambda z'\,y)\;z]\!]_{x:\mathsf{Nat},y:A,z:A} = [\![y]\!]_{x:\mathsf{Nat},y:A,z:A} = \{([]\,, [\alpha]\,, []\,, \alpha);\ \alpha \in |[\![A]\!]|\}\,.$$

One can check that:

$$\begin{array}{rr} & x^{[]} : \mathsf{Nat}, y^{[\alpha]} : A, z^{[]} : A \vdash \mathsf{I}\,(\mathsf{S}\,x)\;(\lambda z'\,y)\;z^{\alpha} : A \\ \text{iff} & x^{[]} : \mathsf{Nat}, y^{[\alpha]} : A \vdash \mathsf{I}\,(\mathsf{S}\,x)\;(\lambda z'\,y)^{([],\alpha)} : A \\ \text{iff} & x^{[]} : \mathsf{Nat} \vdash \mathsf{I}\,(\mathsf{S}\,x)^{([([],\alpha)],[],\alpha)} : A \\ \text{iff} & \vdash \mathsf{I}^{([],[([],\alpha)],[],\alpha)} : A \qquad (*) \end{array}$$

and then $\vdash \mathsf{I}\,\mathsf{O}^{([([],\alpha)],[],\alpha)}$ for all $\alpha \in |[\![A]\!]|$. This contradicts the fact that, by the first equation:

$$[\![\mathsf{I}\,\mathsf{O}]\!] = [\![\lambda y\,\lambda z\,(\mathsf{I}\,\mathsf{O}\,y\,z)]\!] = [\![\lambda y\,\lambda z\,z]\!] = \{([]\,, [\alpha]\,, \alpha);\ \alpha \in |[\![A]\!]|\}$$

because we supposed $|[\![A]\!]| \neq \emptyset$. □

Remark 3. The equivalence $(*)$ holds because $[\![\mathsf{S}]\!] = \mathcal{S}_!$ is linear, hence strict: this reflects the general fact that, if $s \in \mathfrak{F}\,(\mathcal{A} \Rightarrow \mathcal{B})$ contains no $([]\,, \beta)$ then, for all $t \in \mathfrak{F}\,(\mathcal{B} \Rightarrow \mathcal{C})$, $([]\,, \gamma)$ in $t \bullet s$ iff $([]\,, \gamma) \in t$. Such a phenomenon was

already noted by Girard in his interpretation of system T in coherence spaces [6]. His evidence that there was no interpretation of the iteration operator using the linear successor relied on a coherence argument. The previous lemma is stronger: notice that this makes no use of hypotheses on finiteness structures; hence, it holds in the relational model as well, and actually any web based model of linear logic where promotion is defined similarly, as soon as the interpretation of successor is strict.

Lazy integers. In short, strict morphisms cannot produce anything *ex nihilo*; but the successor of any integer should be marked as non-zero, for the iterator to distiguish between both cases. Hence the successor should be affine: similarly to Girard's solution, we will interpret Nat by so-called *lazy* natural numbers. Let $\mathcal{N}_l = (|\mathcal{N}_l|, \mathfrak{P}_{\mathrm{f}}(|\mathcal{N}_l|))$ be such that $|\mathcal{N}_l| = \mathbf{N} \cup \mathbf{N}^{>}$, where $\mathbf{N}^{>}$ is just a disjoint copy of $\mathbf{N}$. The elements of $\mathbf{N}^{>}$ are denoted by $k^{>}$, for $k \in \mathbf{N}$: $n^{>}$ represents a partial integer, not fully determined but *strictly greater than* k. If $\nu \in |\mathcal{N}_l|$, we define ν^+ as $k+1$ if $\nu = k$ and $(k+1)^{>}$ if $\nu = k^{>}$. Then we set $\mathcal{S}_l = \{([], 0^{>})\} \cup \{([\nu], \nu^+)\}$, which is affine. Notice that $\mathcal{O} \in \mathfrak{F}(\mathcal{N}_l)$ and $\mathcal{S}_l \in \mathfrak{F}(\mathcal{N}_l \Rightarrow \mathcal{N}_l)$.

We will show that these allow to provide an interpretation of recursion, hence iteration, in system T: for all finiteness space $\mathcal{A}$, there exists $\mathcal{R}_{\mathcal{A}} \in \mathfrak{F}(\mathcal{N}_l \Rightarrow (\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A})$ such that, in $\Lambda_{\underline{\mathrm{Fin}}}$,

$$y : \mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}, z : \mathcal{N}_l \vdash \mathcal{R}\,\mathcal{O}\,y\,z \simeq z : \mathcal{A}$$

and

$$x : \mathcal{N}_l, y : \mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}, z : \mathcal{N}_l \vdash \mathcal{R}\,(\mathcal{S}_l\,x)\,y\,z \simeq y\,x\,(\mathcal{R}\,x\,y\,z) : \mathcal{A}.$$

4 A recursion operator in finiteness spaces

For all finiteness space $\mathcal{A}$, write $\mathcal{R}ec\mathcal{O}p[\mathcal{A}] = \mathcal{N}_l \Rightarrow (\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}$. We want to introduce a recursion operator $\mathcal{R}_{\mathcal{A}} \in \mathfrak{F}(\mathcal{R}ec\mathcal{O}p[\mathcal{A}])$ intuitively subject to the following definition:

$$\mathcal{R}\,t\,u\,v = \text{match } t \text{ with } \begin{cases} \mathsf{O} \mapsto v \\ \mathsf{S}\,t' \mapsto u\,t'\,(\mathcal{R}\,t'\,u\,v) \end{cases}.$$

This definition is recursive, and a natural means to obtain such an operator is as the fixpoint of:

$$\lambda\mathcal{X}^{\mathcal{R}ec\mathcal{O}p}\,\lambda x^{\mathcal{N}_l}\,\lambda y^{\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}}\,\lambda z^{\mathcal{A}}\left(\text{match } x \text{ with } \begin{cases} \mathsf{O} \mapsto z \\ \mathsf{S}\,x' \mapsto y\,x'\,(\mathcal{X}\,x'\,y\,z) \end{cases}\right). \quad (1)$$

Fixpoints. The cartesian closed category $\underline{\mathrm{Rel}}_!$ is cpo-enriched, the order on morphisms being inclusion. Hence it has fixpoints at all types: for all set A and $f \in \underline{\mathrm{Rel}}_!(A, A)$, the least fixpoint of f is $\bigcup_{k \geq 0} f^k\,\emptyset$, which is an increasing union.

The least fixpoint operator is itself definable as the supremum of finitary approximants, $\mathcal{F}ix_A = \bigcup_{k\geq 0} \mathcal{F}ix_A^{(k)}$, where:

$$\begin{aligned}\mathcal{F}ix_A^{(0)} &= \emptyset\\ \mathcal{F}ix_A^{(k+1)} &= \lambda f\left(f\left(\mathcal{F}ix_A^{(k)}\, f\right)\right)\\ &= \left\{\left(\left[\left(\left[\alpha_1,\ldots,\alpha_n\right],\alpha\right)\right] + \sum_{i=1}^{n}\overline{\varphi}_i,\alpha\right);\ \forall i,\ (\overline{\varphi}_i,\alpha_i)\in\mathcal{F}ix_A^{(k)}\right\}.\end{aligned}$$

Notice indeed that if $\mathcal{A}$ is a finiteness space then, for all k, $\mathcal{F}ix_{\mathcal{A}}^{(k)} = \mathcal{F}ix_{|\mathcal{A}|}^{(k)} \in \mathfrak{F}\left((\mathcal{A}\Rightarrow\mathcal{A})\Rightarrow\mathcal{A}\right)$. The fixpoint, however, is not finitary in general: one can check for instance that $\mathcal{F}ix_{\mathcal{N}_l}\,\mathcal{S}_l = \mathbf{N}^{>} \notin \mathfrak{F}\left(\mathcal{N}_l\right)$ hence $\mathcal{F}ix_{\mathcal{N}_l} \notin \mathfrak{F}\left((\mathcal{N}_l\Rightarrow\mathcal{N}_l)\Rightarrow\mathcal{N}_l\right)$. So our definition of a recursor operator is in two steps: we first introduce the finitary approximants $\mathcal{R}_{\mathcal{A}}^{(k)} \in \mathfrak{F}\left(\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right)$ by $\mathcal{R}_{\mathcal{A}}^{(k)} = \mathcal{S}tep_{\mathcal{A}}^{k}\,\emptyset$, where $\mathcal{S}tep \in \mathfrak{F}\left(\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\Rightarrow\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right)$ is defined as in formula (1); then we prove $\mathcal{R}_{\mathcal{A}} = \bigcup_{k\geq 0}\mathcal{R}_{\mathcal{A}}^{(k)} \in \mathfrak{F}\left(\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right)$.

Pattern matching on lazy integers. We introduce a finitary operator $\mathcal{C}ase$, intuitively defined as:

$$\mathcal{C}ase\, t\, u\, v = \text{match } t \text{ with } \begin{cases}\mathsf{O}\mapsto v\\ \mathsf{S}\,t'\mapsto u\,t'\end{cases}.$$

More formally:

Definition 1. *If $\overline{\nu} = [\nu_1,\ldots,\nu_k] \in \mathcal{M}_{fin}\left(|\mathcal{N}_l|\right)$, we write $\overline{\nu}^{+} = \left[\nu_1^{+},\ldots,\nu_n^{+}\right]$. Then for all finiteness space $\mathcal{A}$, we let $\mathcal{C}ase_{\mathcal{A}} = \left\{\left(\left[0\right],\left[\right],\left[\alpha\right],\alpha\right);\ \alpha\in|\mathcal{A}|\right\} \cup \left\{\left(\left[0^{>}\right]+\overline{\nu}^{+},\left[\left(\overline{\nu},\alpha\right)\right],\left[\right],\alpha\right);\ \overline{\nu}\in\mathcal{M}_{fin}\left(|\mathcal{N}_l|\right)\wedge\alpha\in|\mathcal{A}|\right\}$.*

Lemma 2. *The relation $\mathcal{C}ase$ is finitary: $\mathcal{C}ase_{\mathcal{A}} \in \mathfrak{F}\left(\mathcal{N}_l\Rightarrow(\mathcal{N}_l\Rightarrow\mathcal{A})\Rightarrow\mathcal{A}\Rightarrow\mathcal{A}\right)$. Moreover, $y:\mathcal{N}_l\Rightarrow\mathcal{A}, z:\mathcal{A}\vdash \mathcal{C}ase\,\mathcal{O}\,y\,z \simeq z:\mathcal{A}$ and $x:\mathcal{N}_l, y:\mathcal{N}_l\Rightarrow\mathcal{A}, z:\mathcal{A}\vdash \mathcal{C}ase\left(\mathcal{S}_l\,x\right)y\,z\simeq y\,x:\mathcal{A}$.*

Proof That the equations hold is a routine exercise. We use Remark 1 to prove $\mathcal{C}ase$ is finitary. For all $n\in\mathfrak{F}\left(\mathcal{N}\right)$,

$$\begin{aligned}\mathcal{C}ase_{\mathcal{A}}\,n &\subseteq \left\{\left(\left[\right],\left[\alpha\right],\alpha\right);\ \alpha\in|\mathcal{A}|\right\}\cup\left\{\left(\left[\left(\overline{\nu},\alpha\right)\right],\left[\right],\alpha\right);\ \overline{\nu}\in n^{!}\wedge\alpha\in|\mathcal{A}|\right\}\\ &\subseteq \left(\lambda y^{\mathcal{N}_l\Rightarrow\mathcal{A}}\,\lambda z^{\mathcal{A}}\,z\right)\cup\left(\lambda y^{\mathcal{N}_l\Rightarrow\mathcal{A}}\,\lambda z^{\mathcal{A}}\left(y\,n\right)\right)\end{aligned}$$

and the union of two finitary subsets is finitary. In the reverse direction, we are left to prove that, for all $(\overline{\varphi},\overline{\alpha},\alpha)\in|(\mathcal{N}_l\Rightarrow\mathcal{A})\Rightarrow\mathcal{A}\Rightarrow\mathcal{A}|$, setting $N' = \mathcal{C}ase_{\mathcal{A}}^{\perp}\cdot\{(\overline{\varphi},\overline{\alpha},\alpha)\}$, $n^{!}\cap N'$ is finite; this is immediate because N' is finite (it has at most one element). □

A recursor in $\underline{\mathrm{Rel}}_!$. We now introduce the interpretation of recursor as the fixpoint of the operator (1).

Definition 2. *For all finiteness space* $\mathcal{A}$*, we define*

$$\begin{aligned}\mathcal{S}tep_{\mathcal{A}} &= \lambda\mathcal{X}^{\mathcal{R}ec\mathcal{O}p[\mathcal{A}]}\,\lambda x^{\mathcal{N}_l}\,\lambda y^{\mathcal{N}_l\Rightarrow\mathcal{A}\Rightarrow\mathcal{A}}\,\lambda z^{\mathcal{A}}\left(\mathcal{C}ase_{\mathcal{A}}\,x\left(\lambda x'^{\mathcal{A}}\left(y\,x'\left(\mathcal{X}\,x'\,y\,z\right)\right)\right)z\right)\\ &\in \mathfrak{F}\left(\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\Rightarrow\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right)\end{aligned}$$

and, for all $k\in\mathbf{N}$, $\mathcal{R}_{\mathcal{A}}^{(k)}=\mathcal{S}tep_{\mathcal{A}}^{k}\,\emptyset\in\mathfrak{F}\left(\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right)$.

Lemma 3. *The relations* $\mathcal{R}_{\mathcal{A}}^{(k)}$ *are given by:* $\mathcal{R}_{\mathcal{A}}^{(0)}=\emptyset$ *and*

$$\mathcal{R}_{\mathcal{A}}^{(k+1)}=\left\{\left(\left[0\right],\left[\right],\left[\alpha\right],\alpha\right);\ \alpha\in\left|\mathcal{A}\right|\right\}\cup\left\{\begin{array}{r}\left(\left[0^{>}\right]+\sum_{i=0}^{n}\overline{\nu}_i^{+},\left[\left(\overline{\nu}_0,\left[\alpha_1,\ldots,\alpha_n\right],\alpha\right)\right]+\sum_{i=1}^{n}\overline{\varphi}_i,\sum_{i=1}^{n}\overline{\alpha}_i,\alpha\right);\\ \forall i,\ \left(\overline{\nu}_i,\overline{\varphi}_i,\overline{\alpha}_i,\alpha_i\right)\in\mathcal{R}_{\mathcal{A}}^{(k)}\end{array}\right\}.$$

Proof Routine exercise. □

Definition 3. *We set* $\mathcal{R}_{\mathcal{A}}=\bigcup_{k\geq 0}\mathcal{R}_{\mathcal{A}}^{(k)}\in\left|\mathcal{R}ec\mathcal{O}p\left[\mathcal{A}\right]\right|$.

At that point, we already have a recursor in $\underline{\mathrm{Rel}}_!$:

Theorem 3 (Correctness of the recursion operator). *If, following Remark 2, we consider the relational semantics of typed* λ*-calculi, then in the internal language* $\Lambda_{\underline{\mathrm{Rel}}}$ *of* $\underline{\mathrm{Rel}}_!$*, the following conversions hold:*

$$y:\left|\mathcal{N}_l\right|\to A\to A,z:A\vdash\mathcal{R}\,\mathcal{O}\,y\,z\simeq z:A$$

and

$$x:\left|\mathcal{N}_l\right|,y:\left|\mathcal{N}_l\right|\to A\to A,z:A\vdash\mathcal{R}\left(\mathcal{S}_l\,x\right)y\,z\simeq y\,x\left(\mathcal{R}\,x\,y\,z\right):A.$$

Proof This follows directly from Lemma 2 and the fact that $\mathcal{R}=\mathcal{S}tep\,\mathcal{R}$:

$$\mathcal{R}\,\mathcal{O}\,y\,z\simeq\mathcal{S}tep\,\mathcal{R}\,\mathcal{O}\,y\,z\simeq\mathcal{C}ase\,\mathcal{O}\left(\lambda x'\left(y\,x'\left(\mathcal{R}\,x'\,y\,z\right)\right)\right)z\simeq z$$

and

$$\begin{aligned}\mathcal{R}\left(\mathcal{S}_l\,x\right)y\,z&\simeq\mathcal{S}tep\,\mathcal{R}\left(\mathcal{S}_l\,x\right)y\,z\\ &\simeq\mathcal{C}ase\left(\mathcal{S}_l\,x\right)\left(\lambda x'\left(y\,x'\left(\mathcal{R}\,x'\,y\,z\right)\right)\right)z\\ &\simeq\lambda x'\left(y\,x'\left(\mathcal{R}\,x'\,y\,z\right)\right)x\\ &\simeq y\,x\left(\mathcal{R}\,x\,y\,z\right).\end{aligned}$$

□

Finiteness. It only remains to prove $\mathcal{R}$ is finitary.

Definition 4. *If* $n\in\mathfrak{F}\left(\mathcal{N}_l\right)$*, we set* $\max\left(n\right)=\max\left\{k;\ k\in n\vee k^{>}\in n\right\}$*, with the convention that* $\max\left(\emptyset\right)=0$*; then if* $\overline{\nu}\in\left|!\mathcal{N}_l\right|$ *we set* $\max\left(\overline{\nu}\right)=\max\left(\mathrm{Supp}(\nu)\right)$ *and if* $\overline{n}\in\mathfrak{F}\left(!\mathcal{N}_l\right)$, $\max\left(\overline{n}\right)=\max\left(\bigcup_{\overline{\nu}\in\overline{n}}\mathrm{Supp}(\overline{\nu})\right)$.

Lemma 4. *For all* $\gamma=\left(\overline{\nu},\overline{\varphi},\overline{\alpha},\alpha\right)\in\mathcal{R}_{\mathcal{A}}$, $\gamma\in\mathcal{R}_{\mathcal{A}}^{(\max(\overline{\nu})+1)}$.

Proof By induction on $\max\left(\overline{\nu}\right)$, using Lemma 3. □

Lemma 5. *If* $n \in \mathfrak{F}(\mathcal{N}_l)$*, then* $\mathcal{R}_{\mathcal{A}}\, n \in \mathfrak{F}((\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A})$.

Proof The previous Lemma entails $\mathcal{R}_{\mathcal{A}}\, n = \mathcal{R}_{\mathcal{A}}^{(\max(n)+1)}\, n$. We conclude recalling that $\mathcal{R}_{\mathcal{A}}^{(\max(n)+1)}\, n \in \mathfrak{F}((\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A})$. □

Definition 5. *For all* $\overline{\alpha} = [\alpha_1, \ldots, \alpha_k] \in |!\mathcal{A}|$*, we write* $\#(\overline{\alpha}) = k$ *for its size. For all* $\overline{\varphi} = [(\overline{\nu}_1, \overline{\alpha}_1, \alpha_1), \ldots, (\overline{\nu}_k, \overline{\alpha}_k, \alpha_k)] \in |\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}|$*, we set* $\#\#(\overline{\varphi}) = \sum_{j=1}^{k} \#(\overline{\nu}_j)$.

Lemma 6. *If* $(\overline{\nu}, \overline{\varphi}, \overline{\alpha}, \alpha) \in \mathcal{R}_{\mathcal{A}}$*, then* $\#(\nu) = \#(\overline{\alpha}) + \#(\overline{\varphi}) + \#\#(\overline{\varphi})$.

Proof Using Lemma 3, the result is proved for $(\overline{\nu}, \overline{\varphi}, \overline{\alpha}, \alpha) \in \mathcal{R}_{\mathcal{A}}^{(k)}$, by induction on k. □

Theorem 4 (The recursion operator is finitary). $\mathcal{R}_{\mathcal{A}} \in \mathfrak{F}(\mathcal{R}ec\mathcal{O}p[\mathcal{A}])$.

Proof By Remark 1 and Lemma 5, we are left to prove that, for all $n \in \mathfrak{F}(\mathcal{N}_l)$ and all $\gamma \in |(\mathcal{N}_l \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}|$, $N = n^! \cap \left(\mathcal{R}_{\mathcal{A}}^{\perp} \cdot \{\gamma\}\right)$ is finite. But by Lemma 6,

$$N \subseteq \{\overline{\nu} \in |!\mathcal{N}_l|\,;\ \#(\overline{\nu}) = \#(\overline{\alpha}) + \#(\overline{\varphi}) + \#\#(\overline{\varphi}) \wedge \max(\overline{\nu}) \leq \max(n)\}$$

which is finite. □

Remark 4. We keep calling $\mathcal{R}$ "the" recursion operator, but notice such an operator is not unique in $\underline{\mathrm{Rel}}_!$ or $\underline{\mathrm{Fin}}_!$. Consider $\mathcal{C}ase'_{\mathcal{A}} = \{([0,0], [], [\alpha], \alpha);\ \alpha \in |\mathcal{A}|\} \cup \left\{([0^{>}] + \overline{\nu}^{+}, [(\overline{\nu}, \alpha)], [], \alpha);\ \overline{\nu} \in \mathcal{M}_{\mathrm{fin}}(|\mathcal{N}_l|) \wedge \alpha \in |\mathcal{A}|\right\}$. This variant of matching operator behaves exactly like $\mathcal{C}ase$, and one can reproduce our construction of the recursor based on that.

5 About iteration

We have just provided a semantics of the recursor variant of system T in finiteness spaces. From this, we can deduce the following interpretation of the iteration operator:

Definition 6. *For all finiteness space* $\mathcal{A}$*, we write* $\mathcal{I}t\mathcal{O}p[\mathcal{A}] = \mathcal{N}_l \Rightarrow (\mathcal{A} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{A} \Rightarrow \mathcal{A}$*, and set* $\mathcal{I}_{\mathcal{A}} = \lambda x^{\mathcal{N}_l}\, \lambda y^{\mathcal{A} \Rightarrow \mathcal{A}}\, \lambda z^{\mathcal{A}}\, \left(\mathcal{R}_{\mathcal{A}}\, x\, \left(\lambda x'^{\mathcal{N}_l}\, y\right)\, z\right) \in \mathfrak{F}(\mathcal{I}t\mathcal{O}p[\mathcal{A}])$.

By Theorem 3 this defines an internal implementation of an iteration operator, and we obtain that the triple $(\mathcal{N}_l, \mathcal{O}, \mathcal{S}_l)$ is a weak natural number object in the cartesian closed category $\underline{\mathrm{Fin}}_!$ [4,5]:

Lemma 7. *For all* $f \in \underline{\mathrm{Fin}}_!(\mathcal{A}, \mathcal{A})$*, for all* $a \in \mathfrak{F}(\mathcal{A})$*, there is* $h \in \underline{\mathrm{Fin}}_!(\mathcal{N}_l, \mathcal{A})$ *such that* $h\,\mathcal{O} = a$ *and* $h \bullet \mathcal{S}_l = f \bullet h$.

Proof Take $h = \lambda x^{\mathcal{N}_l}\, (\mathcal{I}_{\mathcal{A}}\, x\, f\, a)$. □

We could also have introduced $\mathcal{I}$ by a construction similar to that of $\mathcal{R}$:

Definition 7. *Let*

$$\begin{aligned}\mathcal{I}t\mathcal{S}tep_{\mathcal{A}} &= \lambda\mathcal{X}^{\mathcal{I}t\mathcal{O}p[\mathcal{A}]}\,\lambda x^{\mathcal{N}_l}\,\lambda y^{\mathcal{A}\Rightarrow\mathcal{A}}\,\lambda z^{\mathcal{A}}\left(\mathcal{C}ase_{\mathcal{A}}\,x\left(\lambda x'^{\mathcal{A}}\left(y\left(\mathcal{X}\,x'\,y\,z\right)\right)\right)z\right)\\ &\in \mathfrak{F}\left(\mathcal{I}t\mathcal{O}p\left[\mathcal{A}\right]\Rightarrow\mathcal{I}t\mathcal{O}p\left[\mathcal{A}\right]\right)\end{aligned}$$

and, for all $k\in\mathbf{N}$, $\mathcal{I}_{\mathcal{A}}^{(k)}=\mathcal{I}t\mathcal{S}tep_{\mathcal{A}}^{k}\,\emptyset\in\mathfrak{F}\left(\mathcal{I}t\mathcal{O}p\left[\mathcal{A}\right]\right)$.

Lemma 8. *The relations* $\mathcal{I}_{\mathcal{A}}^{(k)}$ *are given by:* $\mathcal{I}_{\mathcal{A}}^{(0)}=\emptyset$ *and*

$$\mathcal{I}_{\mathcal{A}}^{(k+1)}=\left\{\left(\left[0\right],\left[\right],\left[\alpha\right],\alpha\right);\ \alpha\in\left|\mathcal{A}\right|\right\}\cup\left\{\begin{array}{r}\left(\left[0^{>}\right]+\sum_{i=1}^{n}\overline{\nu}_i^{+},\left[\left(\left[\alpha_1,\ldots,\alpha_n\right],\alpha\right)\right]+\sum_{i=1}^{n}\overline{\varphi}_i,\sum_{i=1}^{n}\overline{\alpha}_i,\alpha\right);\\ \forall i,\ \left(\overline{\nu}_i,\overline{\varphi}_i,\overline{\alpha}_i,\alpha_i\right)\in\mathcal{I}_{\mathcal{A}}^{(k)}\end{array}\right\}.$$

Proof Again, routine exercise. □

Lemma 9. *We have* $\bigcup_{k\geq 0}\mathcal{I}_{\mathcal{A}}^{(k)}=\mathcal{I}_{\mathcal{A}}$.

Proof Check that, for all k, $\mathcal{I}_{\mathcal{A}}^{(k)}=\lambda x^{\mathcal{N}_l}\,\lambda y^{\mathcal{A}\Rightarrow\mathcal{A}}\,\lambda z^{\mathcal{A}}\left(\mathcal{R}_{\mathcal{A}}^{(k)}\,x\left(\lambda x'^{\mathcal{N}_l}\,y\right)z\right)$. □

A uniformity property of iteration. We can now demonstrate how the concept of recursion is richer than that of iteration in the setting of the finiteness space semantics, or the relational one for that matter. Indeed, one distictive feature of the model is that it is non-uniform. Indeed, if $a,a'\in\mathfrak{F}\left(\mathcal{A}\right)$ then $a\cup a'\in\mathfrak{F}\left(\mathcal{A}\right)$; and in the construction of $a^{!}\in{!}\mathcal{A}$, there is no condition on the elements of the multisets we consider: $a^{!}=\mathcal{M}_{\mathrm{fin}}\left(a\right)$. This is very different from the setting of coherence spaces for instance. But the iterator only considers uniform sets of integers, in the following sense:

Definition 8. *If* $k\in\mathbf{N}$, *we define* $\underline{k}=\mathcal{S}_l^k\,\mathcal{O}=\left\{l^{>};\ l<k\right\}\cup\left\{k\right\}\in\mathfrak{F}\left(\mathcal{N}_l\right)$. *We say* $n\in\mathfrak{F}\left(\mathcal{N}_l\right)$ *is uniform if* $n\subseteq\underline{k}$ *for some* k.

Notice that, in the coherence space of lazy natural numbers used by Girard in [6] to interpret system T, the sets $\underline{k}$ are the finite maximal cliques. Coherence is indeed given by: $k\subset l$ iff $k=l$, $k\subset l^{>}$ iff $k>l$ and $k^{>}\subset l^{>}$ for all k,l. There is also an infinite maximal clique, which is just $\mathbf{N}^{>}$ (recall this was the fixpoint of $\mathcal{S}_l$). We prove $\mathcal{I}$ considers only uniform integers.

Definition 9. *Let be given the following relations in* $\mathfrak{F}\left(\mathcal{I}t\mathcal{O}p\left[\mathcal{A}\right]\right)$: $\mathcal{S}tage_{\mathcal{A}}^{(0)}=\left\{\left(\left[0\right],\left[\right],\left[\alpha\right],\alpha\right);\ \alpha\in\left|\mathcal{A}\right|\right\}$; $\mathcal{S}tage_{\mathcal{A}}^{(1)}=\left\{\left(\left[0^{>}\right],\left[\left(\left[\right],\alpha\right)\right],\left[\right],\alpha\right);\ \alpha\in\left|\mathcal{A}\right|\right\}$; *and, for all* $k>0$, $\mathcal{S}tage_{\mathcal{A}}^{(k+1)}=\mathcal{I}_{\mathcal{A}}^{(k+1)}\setminus\mathcal{I}_{\mathcal{A}}^{(k)}$.

Check that $\mathcal{S}tage_{\mathcal{A}}^{(0)}\cup\mathcal{S}tage_{\mathcal{A}}^{(1)}=\mathcal{I}_{\mathcal{A}}^{(1)}$, hence $\mathcal{I}_{\mathcal{A}}=\bigcup_{k\geq 0}\mathcal{S}tage_{\mathcal{A}}^{(k)}$.

Lemma 10. *Suppose* $\mathcal{A}\neq\mathbf{0}$. *Then, for all* $k\in\mathbf{N}$,

$$\bigcup\left\{\mathrm{Supp}(\overline{\nu});\ \exists(\overline{\varphi},\overline{\alpha},\alpha),\ \left(\overline{\nu},\overline{\varphi},\overline{\alpha},\alpha\right)\in\mathcal{S}tage_{\mathcal{A}}^{(k)}\right\}=\underline{k}.$$

Proof That all the elements are in $\underline{k}$ is easy by induction on k, using Lemma 8. To prove the reverse inclusion, consider $\lambda x^{\mathcal{N}}\,\lambda z^{\mathcal{A}}\left(\mathcal{I}_{\mathcal{A}}\,x\left(\lambda z'^{\mathcal{A}}\,z'\right)z\right)$. □

As a consequence, for all $(\overline{\nu}, \overline{\varphi}, \overline{\alpha}, \alpha) \in \mathcal{I}$, $\mathrm{Supp}(\overline{\nu})$ is uniform. Of course, no such property holds for $\mathcal{R}$, because

$$\mathcal{R}^{(1)}_{\mathcal{A}} \supseteq \left\{ \left(\left[0^{>} \right] + \overline{\nu}^{+}, \left[(\overline{\nu}, [] , \alpha) \right] , [] , \alpha \right) ; \ \alpha \in |\mathcal{A}| \wedge \overline{\nu} \in \mathcal{M}_{\mathrm{fin}} \left(|\mathcal{N}_l| \right) \right\}.$$

An immediate consequence is that no recursor can be derived from $\mathcal{I}$.

Aknowledgements

The present work stems from a discussion with Thomas Ehrhard. It also greatly benefited from many working sessions in the company of Christine Tasson, to whom I am most grateful.

References

1. Ehrhard, T.: Finiteness spaces. Mathematical. Structures in Comp. Sci. **15**(4) (2005) 615–646
2. Ehrhard, T., Regnier, L.: The differential lambda-calculus. Theoretical Computer Science **309** (2003) 1–41
3. Ehrhard, T., Regnier, L.: Differential interaction nets. Electr. Notes Theor. Comput. Sci. **123** (2005) 35–74
4. Thibault, M.F.: Pre-recursive categories. Journal of Pure and Applied Algebra **24** (1982) 79–93
5. Lambek, J., Scott, P.J.: Introduction to higher order categorical logic. Cambridge University Press, New York, NY, USA (1988)
6. Girard, J.Y., Taylor, P., Lafont, Y.: Proofs and types. CUP, Cambridge (1989)
7. Bierman, G.M.: What is a categorical model of intuitionistic linear logic? In Dezani, M., ed.: Proceedings of Conference on Typed lambda calculus and Applications, Springer-Verlag LNCS 902 (1995)